2020

A novel penalty-based wrapper objective function for feature selection in big data using cooperative co-evolution

A.N.M. Bazlur Rashid

Mohiuddin Ahmed
Edith Cowan University, mohiuddin.ahmed@ecu.edu.au

Leslie F. Sikos
Edith Cowan University, l.sikos@ecu.edu.au

Paul Haskell-Dowland
Edith Cowan University, p.haskell-dowland@ecu.edu.au

10.1109/ACCESS.2020.3016679
Rashid, A. B., Ahmed, M., Sikos, L. F., & Haskell-Dowland, P. (2020). A Novel Penalty-Based Wrapper Objective Function for Feature Selection in Big Data Using Cooperative Co-Evolution. *IEEE Access*, *8*, 150113-150129. https://doi.org/10.1109/ACCESS.2020.3016679

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 15, 2020, accepted August 10, 2020, date of publication August 14, 2020, date of current version August 25, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3016679

A Novel Penalty-Based Wrapper Objective Function for Feature Selection in Big Data Using Cooperative Co-Evolution

A. N. M. BAZLUR RASHID, MOHIUDDIN AHMED, LESLIE F. SIKOS, AND PAUL HASKELL-DOWLAND, (Senior Member, IEEE)
School of Science, Edith Cowan University, Perth, WA 6027, Australia

Corresponding author: A. N. M. Bazlur Rashid (a.rashid@ecu.edu.au)

This work was supported in part by the Edith Cowan University (ECU) Higher Degree by Research Scholarship (HDRS), and in part by the ECU School of Science Research Scholarship.

ABSTRACT The rapid progress of modern technologies generates a massive amount of high-throughput data, called Big Data, which provides opportunities to find new insights using machine learning (ML) algorithms. Big Data consist of many features (also called attributes); however, not all these are necessary or relevant, and they may degrade the performance of ML algorithms. Feature selection (FS) is an essential preprocessing step to reduce the dimensionality of a dataset. Evolutionary algorithms (EAs) are widely used search algorithms for FS. Using classification accuracy as the objective function for FS, EAs, such as the cooperative co-evolutionary algorithm (CCEA), achieve higher accuracy, even with a higher number of features. Feature selection has two purposes: reducing the number of features to decrease computations and improving classification accuracy, which are contradictory but can be achieved using a single objective function. For this very purpose, this paper proposes a penalty-based wrapper objective function. This function can be used to evaluate the FS process using CCEA, hence called Cooperative Co-Evolutionary Algorithm-Based Feature Selection (CCEAFS). An experiment was performed using six widely used classifiers on six different datasets from the UCI ML repository with FS and without FS. The experimental results indicate that the proposed objective function is efficient at reducing the number of features in the final feature subset without significantly reducing classification accuracy. Based on different performance measures, in most cases, naïve Bayes outperforms other classifiers when using CCEAFS.

INDEX TERMS Big data, feature selection, cooperative co-evolution, penalty-based wrapper objective function, machine learning.

I. INTRODUCTION

A massive volume of data is continuously generated by modern technologies in a variety of sectors including: healthcare, finance, and economics. This high-throughput data generation is termed Big Data, characterized by large volume, variety, velocity, and veracity. These characteristics correspond to the large amount of data generation, the varying structure and serialization of the data, the speed of data generated, and the accuracy of the data. The availability of large-scale data allows the research community to rapidly discover new knowledge [1]. Several machine learning (ML) algorithms can be applied to big data consisting of large data samples to learn, predict, and classify data for producing accurate results. An increased number of healthcare applications utilizing ML classifiers are studied in the literature. Diagnosis and identifying biomarkers from medical datasets are prominent examples of widely used ML classifiers [2].

There are large number of real-world problems consisting of many features (also called attributes in datasets). However, not all of these features are important as some are irrelevant or redundant, which may result in a lower performance of ML classifiers [3], [4]. Feature selection (FS) is a technique to select the relevant features to reduce data dimensionality, which ultimately improves ML performance [1]. Formally speaking, FS is a process to select a subset of m features from

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

a full set of n features in the dataset by removing irrelevant and non-important features and represent the dataset with a reduced number of features [2]. FS process at first requires a search technique (e.g., greedy search) to discover subsets of features. Next, evaluation measures, such as classification accuracy, are used to evaluate the subsets. A termination condition, such as the number of generations, terminates the FS process. Finally, a validation procedure tests the validity of the selected feature subset [5].

2^k possible solutions for a dataset of n features make computational difficulties for an FS process. With this large search space, a wide range of search algorithms have been applied to the FS process, such as greedy search, best search, and evolutionary search [6]. Among the search techniques, evolutionary algorithms (EA) are best suited to FS processes. However, as the search space increases with the number of data samples and features, the effectiveness of EAs are not satisfactory in most cases. The cooperative co-evolutionary algorithm (CCEA), a meta-heuristic algorithm, applies a divide-and-conquer technique, and is proven to be effective for different applications, including a limited number of FS applications. CCEA decomposes a large and complex problem into several sub-problems, optimizes each sub-problem independently, and collaborates different sub-problems only to build a complete solution of the problem [1].

However, as the objectives of FS are twofold (reducing the number of features, and improving classification accuracy), an appropriate single objective function is required that satisfies the FS objectives [2]. This objective function can be used as the fitness function for the CCEA-based FS (CCEAFS) to converge the algorithm in an attempt to reduce the number of features without significantly decreasing the classification performance of the ML classifiers.

In this paper, an FS process has been studied with the CCEA using six widely used ML classifiers, naïve Bayes (NB) [7], support vector machine (SVM) [8], k-Nearest Neighbour (k-NN) [9], J48 [10], random forest (RF) [11], and logistic regression (LR) [12] on six different datasets from the *UCI ML repository*.[1] At first, the ML classifiers have been applied to all datasets without reducing the dimensions. Next, CCEAFS is applied to all classifiers to reduce the dimensions of the datasets. To support the CCEA search process and satisfy the objectives of the FS process, a penalty-based wrapper objective function has been proposed, which is used in CCEAFS as the fitness function. The comparative results have been analyzed based on different performance matrices, including precision, recall, F1 score, accuracy, micro-averaged, macro-averaged, and weighted averaged precision, recall, and F1 score [13].

The contributions of the paper are as follows:

- presenting a systematic literature review on CCEA-based FS approaches;
- investigating the application of cooperative co-evolution for the feature selection problem;
- proposing a new penalty-based wrapper objective function for feature selection process using cooperative co-evolution;
- investigating the performance of six ML classifiers on six datasets with and without using feature selection;
- proving that the feature selection process does not degrade the performance of the classifiers to a significant amount;
- analysing the effect of the feature selection process on different datasets of a higher number of samples and a lower number of features.

The rest of the paper is organized as follows: Section II presents the literature review on feature selection using cooperative co-evolution. Section III includes the proposed feature selection approach and review on cooperative co-evolution technique. The proposed penalty-based wrapper objective function is illustrated in Section IV. The experimental results are presented and analyzed in Section V. The conclusion and future work directions are included in Section VI.

II. LITERATURE REVIEW

In this section, a review of the feature selection techniques using cooperative co-evolution technique is presented.

Many FS approaches studied in the literature are based on different metrics, including information theory, probability distribution, or classification accuracy [14]. Table 1 presents a taxonomy of different FS approaches.

TABLE 1. Taxonomy of FS approaches [1], [15], [16].

FS approaches	Evaluation methods	Examples of evaluator
Evaluation criteria	Filter [17]	T-test [18], information theory [19], Distributed FS using SC [20]
	Wrapper [21]	SVM [22], k-NN [23]
	Embedded [24]	LASSO [25], Gradient boosting [26]
Evolutionary computation	EA [27]	GA [28], parallel GA [29], GP [30]
	CEA [31]	CCEA [32]
	Swarm optimization [28]	PSO [28], ACO [33]
	Hybrid	TLBO+GSA [34], CMIM+BGA [35], mRMR-TLBOL [36]
	Others	DE [37], MA [38], ABC [37]
Number of objectives	Single-objective [39]	GA [28]
	Multi-objectives [40]	Nondominated sorting GA-II [41]

Note: GP–Genetic Programming, PSO–Particle Swarm Optimization, ACO–Ant Colony Optimization, TLBO–Teaching Learning-Based Algorithm, GSA–Gravitational Search Algorithm, CMIM–Conditional Mutual Information Maximization, BGA–Binary Genetic Algorithm, mRMR–Minimum Redundancy Maximum Relevance, TLBOL–TLBO With Opposition–Based Learning, DE–Differential Evolution, MA–Memetic Algorithm, LCS–Learning Classifier System, ES–Evolutionary Strategy, ABC–Artificial Bee Colony.

In the literature, there is a limited number of FS research works based on CCEA. One of them is a pedestrian detection system using a sub-population size adjustment technique to manage the feature proportion, and this method performed

[1] http://archive.ics.uci.edu/ml/

better with a genetic algorithm (GA), greedy approaches, and random selection [42]. The maximum number of image features used in the experiment was 400. This work has been reproduced by [43] with the varying experimental environment and with more negative samples; however, they achieved comparable results only.

In 2009, Derrac *et al.* proposed a GA-based CCEA for the combined use of instance selection (IS) and feature selection (FS) using three sub-populations (representing IS, FS, and IS and FS together) [44]. Because IS and FS are performed in a single process, this approach takes fewer computations; however, it requires the verification of datasets with noisy instances and a large number of features. A year later, the same research group proposed another IFS method based on the previous concept of the three sub-populations and the use of a k-NN classifier [45]. The used wide-ranging datasets with a higher number of samples and a lower number of features for the experiments, which resulted in improved performance; however, it still required verification for datasets having a greater number of features compared to a lower number of samples.

A dual population-based CCEA was proposed by Tian *et al.* in 2010 [31] for FS and network identification to train the radial bias function neural network (RBFNN) on a range of 26 real-world classification problems. There were a maximum of 20,000 samples, 180 features, and 26 classes. In terms of better accuracy and a reduced number of features to tackle multi-objective optimization, the proposed method performs better.

A CCEA-based embedded FS approach proposed in [46] used learning classifier systems (LCSs) with MA as a local search to improve the performance of LCS for evaluating the fitness of the selected feature subset. 11 benchmark binary class datasets from the UCI repository with a higher number of samples and a lower number of features were used to perform the experiment. Furthermore, the Wilcoxson paired signed ranks test and a non-parametric pairwise statistical test were used to validate the performance of the proposed approach.

In 2018, Ebrahimpour *et al.* proposed an FS technique based on CCEA by dividing datasets vertically in a random fashion and using BGSA (binary gravitational search algorithm) for each solution space [47]. They used information gain weights and Pearson correlation coefficients for evaluating the fitness function. Seven binary microarray datasets with a large number of features and a low number of samples were used to perform the experiment, and achieved better results compared to other methods.

A framework for a clinical decision support system (CDSS) was proposed, which used cooperative co-evolution [48]. The proposed framework considers FS and IS as independent sub-problems and used a wrapper-based approach for FS and IS, where random forest classifier evaluates the selected feature subset. Seven clinical datasets from the UCI ML repository have been used to evaluate the proposed approach and achieved the highest classification accuracy in most cases compared to the state-of-the-art techniques. Table 2 presents a summary of the papers reviewed above with key features.

From the literature review of the FS based on CCEA, it is observed that CCEA is an emerging area of research and only a limited number of applications are available. Among the papers reviewed, the majority of the papers focused on FS and IS together; only a couple of papers addressed the FS problem. The performance of the CCEA mostly depends on the decomposition methods, optimizers, and collaboration techniques [50]. Hence, CCEA based optimizations are still unexplored in many areas and need to be investigated. Based on this literature review, there is no extensive research performed on these techniques using a range of datasets yet. In addition, studies from [16], [51], [52] show that evolutionary computations mostly use algorithms for complex and large optimization problems, such as the FS problem for big data. To address these issues, a CCEA-based FS approach is proposed in the next section.

III. A NOVEL FEATURE SELECTION APPROACH

This section first describes CCEA, then illustrates the methodology of the proposed FS approach based on CCEA.

A. COOPERATIVE CO-EVOLUTION

Potter and De Jong first introduced the cooperative co-evolutionary approach in 1994 for solving large-scale, complex optimization problems [53]. They used a divide-and-conquer approach to split a large problem into several sub-problems, and evolved the interacting co-adapted sub-problems to build a complete solution. The general architecture of the cooperative co-evolutionary algorithm (CCEA) is illustrated in [1]. Examples of optimizing real-world problems with promising performance by CCEA includes function optimization [53], designing artificial neural networks [54], and machine learning applications [55]. A CCEA is comprised of three fundamental phases: 1) problem decomposition, 2) sub-problems evolution, and 3) collaboration and evaluation, as shown in Fig. 1. A brief description of each phase is presented in the following section.

1) PROBLEM DECOMPOSITION

The first phase of the CCEA is to decompose a large problem into multiple sub-problems, which usually depends on the problem structure [50]. The problem decomposition techniques can be static or dynamic.

Consider the function, $y = f(x_1, x_2, \ldots, x_n)$. It can be decomposed into $y_1, y_2, \ldots, y_m$, where $y_1 = f_1(x_1, x_2, \ldots, x_n)$, $y_2 = f_2(x_1, x_2, \ldots, x_n)$, $\ldots y_m = f_m(x_1, x_2, \ldots, x_n)$.

If the function is decomposed statically to have one variable for each sub-problem, then $y_1 = f_1(x_1)$, $y_2 = f_2(x_2), \ldots, y_m = f_m(x_n)$. In contrast, if the function is decomposed dynamically, the grouping of variables into sub-problems will be different than with static decomposition.

TABLE 2. Summary of the papers reviewed [1].

References	Methods used	Key features	Limitations
[42], [43]	CEA, two sub-populations for two features group, sub-population size adjustment	An approach of a pedestrian detection system to determine whether a candidate region contains a pedestrian or not.	It detects features, however, does not reduce any feature. The false positive rate is higher than that of the AdaBoost FS algorithm if used for pedestrian detection.
[44], [45]	CCEA, wrapper-based FS, three sub-populations, CHC, SSGA, CGA, PBIL, 1-NN, k-NN, majority voting	Proposed an approach to reduce attributes based on both the instance and feature selection.	Scalability issues, i.e., the proposed approach needs a verification on large datasets with a massive number of features, including noisy and redundant instances.
[31]	CCEA, two sub-populations, ranked-based selection, multilayer perception network (MLP), decaying radius selection clustering (DRSC), Pareto optimal	A hybrid learning algorithm for simultaneous feature selection and network identification using a compact RBFNN model. Here, the first sub-population corresponds to feature selection and the second one represents the RBFNN architecture.	The proposed method requires a more significant number of hidden nodes compared to the GA-based implementation. Testing accuracy decreases if any of the five objectives is removed.
[46]	CCEA, embedded FS, Pittsburgh LCS, MA, Wilcoxon signed-rank test, non-parametric statistical test	An embedded FS for the datasets with more samples and fewer features, where MA is used as a local search to improve the performance of LCS. Furthermore, the performance has been validated using statistical tests.	The influence of the population size representing feature selection and classification is not incorporated, which could improve algorithm convergence.
[47]	CCEA, random vertical decomposition, global search using binary gravitational search algorithm (BGSA), information gain, Pearson correlation coefficients, C4.5, naïve Bayes	A framework to reduce attributes from high-dimensional microarray datasets with a small sample size and a large number of features.	The performance of the proposed method is compared with other approaches by directly extracting results from [49]. In addition, the experimentation was performed with decision tree and naïve Bayes, which requires further verification by other ML classifiers.
[48]	CCEA, random forest, wrapper-based FS	A clinical decision support system for FS and IS considering FS and IS as independent sub-problems.	Only a random forest classifier is used to evaluate the performance of the proposed approach, which needs to be verified by other classifiers. Moreover, accuracy alone as a fitness function is not sufficient because a fitness function has to deal with contradictory FS objectives to reduce the number of features while increasing the classification accuracy.

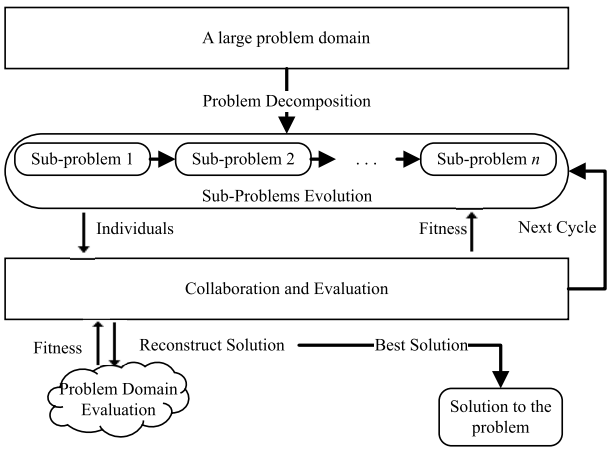

FIGURE 1. CCEA phases [50].

In the case of static decomposition, the problem is decomposed into sub-problems before the evolutionary process starts, and all sub-problems are fixed [56]. In contrast, in the case of dynamic decomposition, a problem is decomposed at the beginning; however, at the time of the evolutionary process, sub-problems can self-adaptively tune to appropriate collaboration levels [57]. A few examples of decomposition techniques are presented in [57]–[59].

2) SUB-PROBLEM EVOLUTION

After the decomposition phase, sub-problems are assigned to different sub-populations, which are then optimized independently by either the same or a different evolutionary optimizer [50]. Sub-problem optimizations can be performed either sequentially or in parallel. Only one sub-population evolves per generation in the former [60], whereas all sub-populations are evolved per generation concurrently in the later case [61]. The most widely used evolutionary optimizer in this area is a genetic algorithm (GA), whereas differential evolution (DE) [62] is the most effective optimizer for CCEA.

3) COLLABORATION AND EVALUATION

Once sub-problems are optimized, the next phase is to interact with different sub-populations to build a complete solution to the problem. The fitness of an individual is evaluated by selecting a collaborator from each sub-population. The performance of the collaboration is assigned as a fitness value to the individual being evaluated. At the end of a CCEA process, individuals with the best collaborations are combined to find the final solution to the problem [50]. 1+1 collaboration [63], the 1+N collaboration model [56], and reference sharing (RS) [50] are examples of collaboration models used in CCEA.

B. METHODOLOGY

In this paper, a novel CCEA-based feature selection approach for Big Data is introduced, called CCEAFS. The proposed methodology of CCEAFS is displayed in Fig. 2.

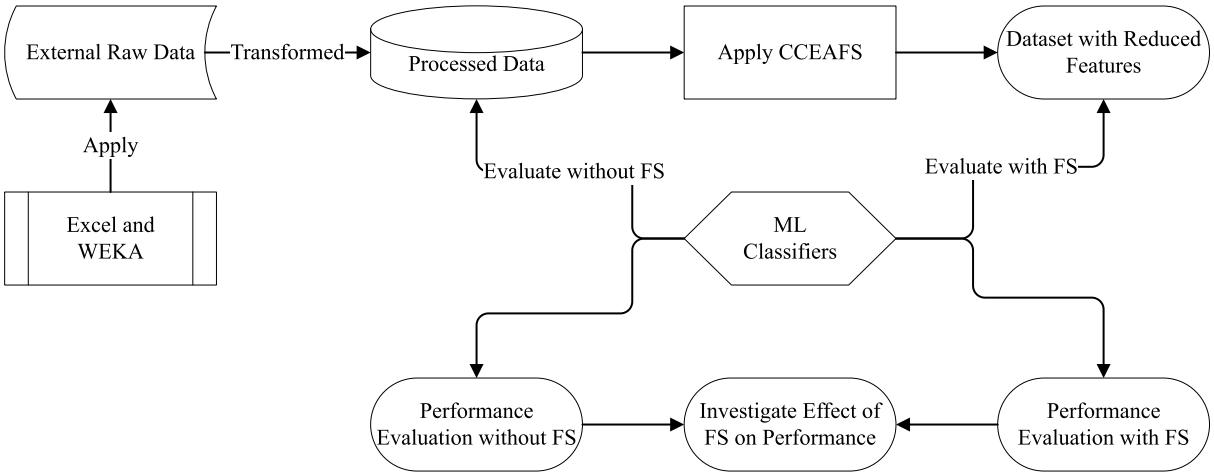

FIGURE 2. Proposed CCEA based FS.

Datasets from the UCI ML repository were collected, and preprocessed using Microsoft Excel and *WEKA*.[2] These datasets were processed using six ML classifiers, NB, SVM, k-NN, J48, RF, and LR. The performance of these classifiers was evaluated based on precision, recall, F1 score, and accuracy. CCEAFS was applied to the datasets to reduce the number of features. The datasets with reduced dimensionality were then processed using the same six ML classifiers, and the performance of each classifier was evaluated based on the aforementioned metrics. The performance results obtained by the classifiers with and without FS were analyzed, and the effect of FS on the performance of the classifiers was investigated. Formally, the proposed CCEAFS can be described as follows.

Assume a dataset consisting of n features:

$$D = f_1, f_2, f_3, \ldots, f_n \tag{1}$$

Dataset D is decomposed into m sub-datasets with n/m features in each sub-dataset:

$$D_1 = \{f_1, f_2, \ldots, f_g\}, D_2 = \{f_1, f_2, \ldots, f_g\}, \\ D_g = \{f_1, f_2, \ldots, f_m\} \tag{2}$$

Each sub-dataset is represented by a sub-population in CCEA. Here, g is the number of genes in each individual and equals to n/m. Consider the size of each sub-population (sp) is s. An example of sub-population sp_1 consisting s individual can be the following:

$$ind_1 = \{0, 1, 1, 0, \ldots, 1\}, ind_2 = \{1, 1, 1, 0, \ldots, 0\}, \\ ind_g = \{0, 1, 1, 1, \ldots, 1\} \tag{3}$$

If a feature is selected in the individual, then it is represented as 1; otherwise it is 0, i.e., the feature is not selected. To evaluate an individual ind_1 in sub-population sp_1, consider collaborators from other sub-populations (ind_2 from sp_2 and ind_4 from sp_3) selected to build a complete solution with reduced features. If static decomposition of twelve features into three sub-populations (each having four features) is assumed, and if $sp_1\{ind_1\} = \{f_1, f_3, f_4\}$, $sp_2\{ind_2\} = \{f_6, f_7, f_8\}$, and $sp_3\{ind_4\} = \{f_9, f_12\}$, then the complete solution is defined as follows:

$$solution = \{f_1, f_3, f_4, f_6, f_7, f_8, f_9, f_{12}\} \tag{4}$$

The solution with this reduced number of features is then sent to the classifiers to measure accuracy and other metrics. The best individual with a reduced number of features and the highest classification accuracy survives the iterations. The best individuals from other sub-populations are used to collaborators from generation 1 onwards. The process continues until it reaches a fixed number of generations, or until no better fitness is achieved over the generations. Algorithm 1 is the pseudocode of the proposed CCEAFS framework. A JAVA-based implementation of the framework is available at GitHub.[3] In the next section, a new penalty-based objective function is proposed to be used as the fitness function for CCEAFS.

IV. A NOVEL PENALTY-BASED OBJECTIVE FUNCTION

Feature selection problem has two purposes: reducing the number of features of the dataset to lower computational cost, while maximizing the classification accuracy to increase the performance of the classification model. However, two-fold objectives are somewhat contradictory. Hence, classification accuracy is not sufficient to evaluate the fitness function of the optimization algorithm for obtaining an optimally reduced feature subset. In the literature, the two objectives are combined into a single objective function for such a problem [2], [45], [64]. The objective function is defined using different variables, including the number of correctly classified instances, the total number of test samples, the number of

[2]https://www.cs.waikato.ac.nz/ml/weka/

[3]https://github.com/bazlurrashid/cooperative-coevolution/tree/CCEAFS/

Algorithm 1 CCEAFS

```
Read the dataset to get the number of features f;
Initialize subPop, subPopSize, generation;
Calculate the length of individual: l = f/subPop;
Statically (starting from the feature indexed at 1) decompose f features into subPop having l features each;
generation = 1;
for x = 1 to subPop do
  for y = 1 to subPopSize do
    Initialize individual with 0 and 1;
  end for
end for
for x = 1 to subPop do
  for y = 1 to subPopSize do
    Find random collaborators for each individual;
  end for
end for
for x = 1 to subPop do
  for y = 1 to subPopSize do
    Evaluate all individuals according to (5) and sort them in descending order of fitness;
  end for
end for
for x = 1 to subPop do
  Find the best individual from each sub-population and sort them in descending order of fitness;
end for
Pick the globally best solution and store the optimal feature subset with the highest fitness value;
while generation ≤ generation_max do
  generation = generation + 1;
  for x = 1 to subPop do
    Evolve each sub-population using a genetic algorithm;
  end for
  for x = 1 to subPop do
    for y = 1 to subPopSize do
      Find the best individuals from the previous generation as collaborators for each individual in the current generation;
    end for
  end for
  for x = 1 to subPop do
    for y = 1 to subPopSize do
      Evaluate all individuals according to (5) and sort them in descending order of fitness;
    end for
  end for
  for x = 1 to subPop do
    Find the best individual from each sub-population and sort them in descending order of fitness;
  end for
  Pick the globally best solution and store the optimal feature subset with the highest fitness value;
end while
```

features selected in the subset, the total number of features in the dataset, and penalty terms. In this paper, a penalty-based wrapper objective function is proposed by combining the two objectives for the feature selection problem for big data using the cooperative co-evolution technique, which is used here as the fitness function. The new objective function is defined as:

$$f = w_1 * f_1 - w_2 * f_2 \tag{5}$$

$$f_1 = T_c/T \tag{6}$$

$$f_2 = S/N \tag{7}$$

where

T_c is the number of correctly classified instances in the test or training samples;

T is the total number of test or training samples in the dataset (the test or training samples depend on the classification mode of using cross-validation or the supplied test set);

S is the number of features selected in the subset;

N is the total number of features in the dataset;

w_1 and w_2 are two control parameters for the objective functions f_1 and f_2, which are used to adjust the penalty term for f_1 and f_2, with $w_1 + w_2 = 1$; and

f is the overall objective function.

Since f is the aggregation of f_1 and f_2, corresponding weightings w_1 and w_2 are associated with f_1 and f_2, respectively. w_1 and w_2 would affect the search optimization process to find optimal individuals (solutions to the problem). Therefore, an empirical experiment can be conducted to find the appropriate values of w_1 and w_2 for a particular problem.

V. RESULTS AND DISCUSSIONS

The experimentation has been performed on six datasets, which have been collected from the publicly available UCI machine library repository.

A. DATASET DETAILS

The datasets used in the experimentation are listed in Table 3. The six different datasets have been used with increasing complexities. The datasets have been selected with a dimensionality from 8 to 1,024 and samples from 170 to 8,992.

TABLE 3. The datasets used for the experiments.

Name	Classes	Features	Samples
Breast cancer Wisconsin[4]	2	30	569
Dermatology[5]	6	34	366
Divorce[6]	2	54	170
Diabetes[7]	2	8	768
Musk[8]	2	166	6,598
QSAR Oral Toxicity[9]	2	1,024	8,992

The most common cancer in women is breast cancer, resulting in many deaths worldwide. The Wisconsin breast cancer dataset contains 357 benign and 212 malignant classes. The features in the dataset include real values of radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension for each cell nucleus [65].

The dermatology dataset contains six types of diseases: psoriasis, seboreic dermatitis, lichen planus, pityriasis rosea, cronic dermatitis, and pityriasis rubra pilaris. It has 12 clinical features and 22 histopathological features. The differential diagnosis of erythemato-squamous diseases in dermatology is problematic, because they share the clinical features of erythema. Furthermore, a disease can show symptoms of other diseases [66].

The divorce dataset contains a number of features measured using the divorce predictors scale based on Gottman couples therapy. 86 samples are married couples, and 84 samples represent divorced people, indicated by a class value of 0 and 1, respectively. The purpose of the dataset is to help family counsellors and family therapists with case formulation and the preparation of an intervention plan [67].

The diabetes dataset contains samples collected from female patients of Pima Indian heritage aged 21 years or over. The predictor variables in this dataset include the number of times the patients have been pregnant, their plasma glucose, diastolic blood pressure, triceps skinfold thickness, insulin level, body mass index, diabetes pedigree function, and age [68].

The musk dataset describes a set of 102 molecules, of which 39 have been annotated by human experts as musks, and 63 as not being musks. These can be used to predict whether new molecules are musks. The dataset has 5,581 musk samples and 1,017 non-musk samples. The stored features include ID, molecule names, conformation name, f1 to f162 as distance features, and f163–f166 are OXY-DIS, OXY-X, OXY-Y, and OXY-Z [69].

The QSAR Oral Toxicity dataset consists of 1,024 molecular fingerprints with binary values and 8,992 samples of chemicals divided into 2 classes: very toxic/positive and not very toxic/negative. Among the chemicals, 741 chemical samples are classified as very toxic/positive and 8,251 chemical samples are classified as not very toxic/negative. The objective of the dataset is to predict very toxic (LD_{50} lower than 50 mg/kg) and nontoxic (LD_{50} greater than or equal to 2,000 mg/kg) endpoints [70].

B. CCEA PARAMETERS

The CCEA parameters that have been used in the experimentation and are common to all of the datasets used are listed in Table 4.

TABLE 4. CCEA parameters details.

Phases	Options
Problem decomposition	Static
Sub-problem evolution	GA
Collaboration and evaluation	1+N

Static decomposition with a variable number of partitions based on the number of features in the dataset has been used. For the Wisconsin breast cancer and divorce datasets, three sub-populations have been used, whereas for the dermatology, diabetes, and musk datasets, two sub-populations. Sub-population size has been kept as 50 for all datasets. In the case of GA optimization, the binary representation of the population is used, in which a binary 1 indicates that a feature is selected and a binary 0 indicates that a feature is not selected from the dataset. Sub-populations were initialized randomly at generation 0. Tournament selection was used to select parent individuals and genetic operators, i.e., by cross-over (cross-over rate = 100%) and mutation (mutation rate = 15%) were used to populate next-generation individuals. Elitism = 1 was used to keep the best individuals to the subsequent generations. In generation 0, since there is no previous history, to evaluate an individual in a sub-population, random collaboration was performed to collaborate with individuals from other sub-populations. In the subsequent generations, the best individuals from the previous generation were used as the collaborators for evaluating an individual in a sub-population. Collaboration performance, i.e., the fitness value was assigned to the individual being evaluated. The best individuals were combined from all sub-populations to obtain the best individual in a generation.

C. EVALUATION METRICS OF THE MODEL

Evaluation metrics measure the quality of the machine learning model. The evaluation metrics, which are used in this article are accuracy, precision, recall, F1 score, micro-averaged precision, micro-averaged recall, micro-averaged F1 score, macro-averaged precision, macro-averaged recall, macro-averaged F1 score, weighted-precision, weighted-recall, and weighted-F1 score [71]–[73].

D. PERFORMANCE EVALUATION OF THE PROPOSED OBJECTIVE FUNCTION

The proposed objective function is used in conjunction with CCEA for finding an optimal subset of features. The performance of the proposed objective function is compared using six different datasets of large sample and a low number of feature characteristics with six classification algorithms, and achieved better results of convergence in each case than the state of the art. The objective function has been tested with different values of penalty terms w_1 and w_2, and $w_1 = 0.60$ and $w_2 = 0.40$ were shown to be effective to converge the CCEAFS with stable classification accuracy and feature number for all of the datasets tested. Fig. 3-7 show CCEAFS convergence on the six datasets.

In most cases, convergence was achieved through a termination condition, i.e., after a fixed number of generations, or until there was no more significant improvement in the fitness value (to avoid computational overhead, the iteration was terminated after 50 successive generations with no improvement). The algorithm was allowed to terminate with 30% of the total number of input generation being successful, if there was no change in the fitness value (occurred in some of the cases only).

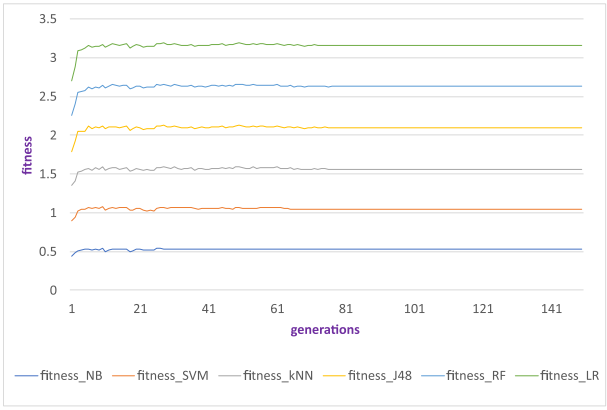

FIGURE 3. Objective function convergence of the Wisconsin breast cancer dataset.

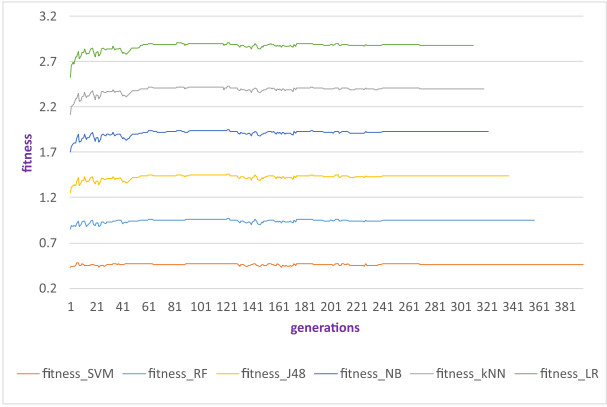

FIGURE 4. Objective function convergence of the dermatology dataset.

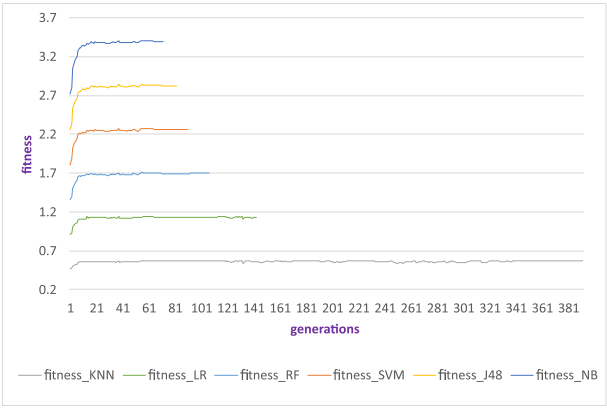

FIGURE 5. Objective function convergence of the divorce dataset.

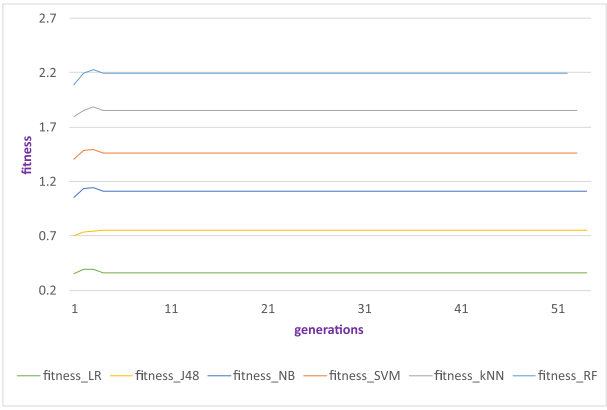

FIGURE 6. Objective function convergence of the diabetes dataset.

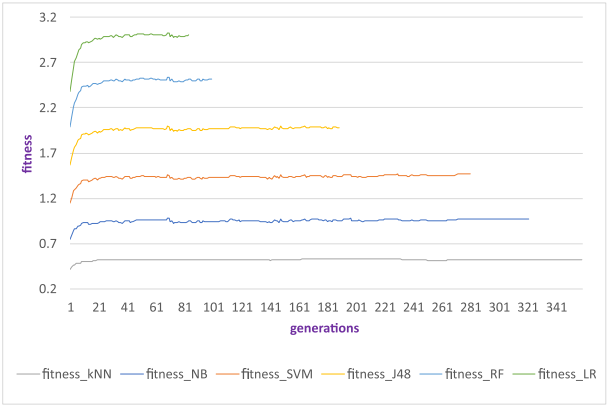

FIGURE 7. Objective function convergence of the musk dataset.

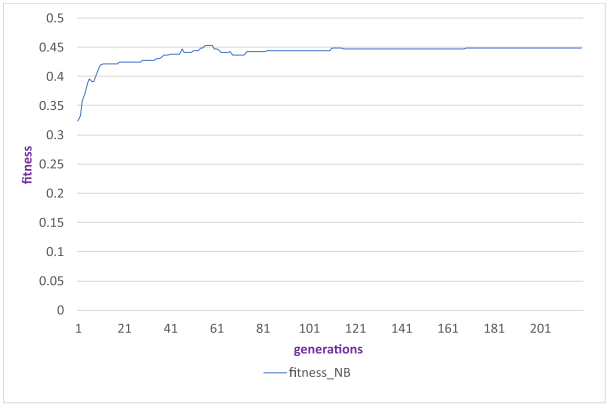

FIGURE 8. Objective function convergence of the QSAR oral toxicity dataset for NB classifier.

E. PERFORMANCE EVALUATION OF CLASSIFIERS WITH CCEA

For the experiments, six widely-used classifiers NB, SVM, *k*-NN, J48, RF, and LR have been used [74]–[76]. First, the datasets have been tested with these classifiers without dimensionality reduction. Second, CCEA has been used to reduce the dataset dimension, and all six classifiers have been used to evaluate the performance of dimensionality reduction, i.e., feature selection. The performance results have been obtained using cross-validation. Table 5 shows the confusion matrices and Table 6 shows detailed accuracy by class and summary results using all classifiers with and without FS for the breast cancer Wisconsin dataset.

From the confusion matrices and detailed accuracy in Table 5 and 6, it can be observed that with all features in the dataset, SVM, *k*-NN, RF, and LR classifiers perform slightly

TABLE 5. Confusion matrices of the wisconsin breast cancer dataset for different classifiers.

Classifiers	TP	FP	FN	TN
NB	189	23	18	339
NB+CCEAFS	**191**	**21**	18	339
SVM	201	11	2	355
SVM+CCEAFS	179	33	8	349
k-NN	200	12	14	343
k-NN+CCEAFS	189	23	20	337
J48	193	19	21	336
J48+CCEAFS	174	38	**13**	**344**
RF	198	14	8	349
RF+CCEAFS	190	22	16	341
LR	201	11	18	339
LR+CCEAFS	187	25	18	339

better than NB, and J48 classifiers in terms of precision, recall, and the F1 score. In the next phase, CCEAFS was applied to all these classifiers to reduce the number of features in the dataset. The number of features in the dataset is 30, which was reduced to 1 using J48+CCEAFS. The reduced number of features using NB+CCEAFS, RF+CCEAFS, and LR+CCEAFS was 2, whereas using SVM+CCEAFS and k-NN+CCEAFS, it was 3. The confusion matrices of these combinations to reduce the number of features from Table 6 indicate that SVM, k-NN, and LR perform better than other classifiers except NB in terms of the same performance measures. When NB is combined with CCEAFS, it performs better and slightly less than RF; however, combining CCEAFS with other classifiers actually reduced the number of features in the dataset without significant reduction of performance measures in terms of precision, recall, and F1 score.

The performance of all classifiers based on accuracy, precision, recall, F1 score, and features on the Wisconsin breast cancer dataset is shown in Fig. 9. Simulation results from Fig. 9 shows that all the classifiers are equally good without using FS and when combined with CCEAFS using these performance measures. The results from Fig. 9 indicate that SVM, k-NN, and RF outperform NB, J48, and LR in terms

TABLE 6. Detailed accuracy by class of the wisconsin breast cancer dataset.

	precision		recall		F1 score	
	NB	**NB+CCEAFS**	**NB**	**NB+CCEAFS**	**NB**	**NB+CCEAFS**
class-M	0.913	**0.914**	0.892	**0.901**	0.902	**0.907**
class-B	0.936	**0.942**	0.950	0.950	0.943	**0.946**
micro-average	0.923	**0.928**	0.921	**0.925**	0.923	**0.926**
macro-average	0.928	**0.931**	0.928	**0.931**	0.928	**0.931**
weighted-average	0.928	**0.931**	0.928	**0.931**	0.928	**0.931**
	precision		**recall**		**F1 score**	
	SVM	**SVM+CCEAFS**	**SVM**	**SVM+CCEAFS**	**SVM**	**SVM+CCEAFS**
class-M	0.990	0.957	0.948	0.844	0.969	0.897
class-B	0.970	0.914	0.994	0.978	0.982	0.945
micro-average	0.980	0.935	0.971	0.911	0.975	0.921
macro-average	0.977	0.930	0.977	0.928	0.977	0.927
weighted-average	0.977	0.928	0.977	0.928	0.977	0.928
	precision		**recall**		**F1 score**	
	k-NN	**k-NN+CCEAFS**	**k-NN**	**k-NN+CCEAFS**	**k-NN**	**k-NN+CCEAFS**
class-M	0.935	0.904	0.943	0.892	0.939	0.898
class-B	0.966	0.936	0.961	0.944	0.963	0.940
micro-average	0.950	0.920	0.952	0.918	0.951	0.919
macro-average	0.954	0.924	0.954	0.924	0.954	0.924
weighted-average	0.954	0.924	0.954	0.924	0.954	0.924
	precision		**recall**		**F1 score**	
	J48	**J48+CCEAFS**	**J48**	**J48+CCEAFS**	**J48**	**J48+CCEAFS**
class-M	0.902	**0.930**	0.910	0.821	0.906	0.872
class-B	0.946	0.901	0.941	**0.964**	0.944	0.931
micro-average	0.924	0.916	0.926	0.892	0.925	0.902
macro-average	0.930	0.912	0.930	0.910	0.930	0.909
weighted-average	0.930	0.910	0.930	0.910	0.930	0.910
	precision		**recall**		**F1 score**	
	RF	**RF+CCEAFS**	**RF**	**RF+CCEAFS**	**RF**	**RF+CCEAFS**
class-M	0.961	0.922	0.934	0.896	0.947	0.909
class-B	0.961	0.939	0.978	0.955	0.969	0.947
micro-average	0.961	0.930	0.956	0.926	0.958	0.928
macro-average	0.961	0.933	0.961	0.933	0.961	0.933
weighted-average	0.961	0.933	0.961	0.933	0.961	0.933
	precision		**recall**		**F1 score**	
	LR	**LR+CCEAFS**	**LR**	**LR+CCEAFS**	**LR**	**LR+CCEAFS**
class-M	0.918	0.912	0.948	0.882	0.933	0.897
class-B	0.969	0.931	0.950	0.950	0.959	0.940
micro-average	0.943	0.922	0.949	0.916	0.946	0.919
macro-average	0.950	0.924	0.949	0.924	0.949	0.924
weighted-average	0.949	0.924	0.949	0.924	0.949	0.924

FIGURE 9. Performance evaluation of classifiers on the Wisconsin breast cancer dataset.

of accuracy, precision, recall, and F1 score. NB, J48, and LR have equally well results in terms of these measures. All these classifiers, when combined with CCEAFS, achieved similar improvements. In other words, NB, RF, and k-NN outperform the other classifiers, and the rest of the classifiers have equally good results. However, with an exception in NB, a reduction in all measures is observed for the rest of the classifiers when combined with CCEAFS.

The performance of all classifiers on the dermatology dataset based on accuracy, precision, recall, F1 score, and features is shown in Fig. 10. Simulation results show that all the classifiers perform equally well without using FS and when combined with CCEAFS, in terms of accuracy, precision, recall, and F1 score. When CCEAFS is applied to all classifiers, the number of features was reduced by NB, J48, RF, and LR from 34 to 7, and by SVM, and k-NN to 8. It can be observed that NB, SVM, RF, and LR perform equally better than other classifiers without using FS. When the classifiers are combined with CCEAFS, NB, k-NN, and J48 perform equally good compared to other classifiers. Accuracy, precision, recall, and the F1 score are comparatively similar in the case of k-NN and a slight reduction can be observed for NB, J48, and RF. However, these performances are reduced in the case of SVM and LR although a huge reduction in the number of features can be achieved using CCEAFS.

The performance of the classifiers based on accuracy, precision, recall, F1 score, and features on the divorce dataset is displayed in Fig. 11. According to the simulation, all classifiers perform equally well without using FS and with CCEAFS. The number of features was reduced to only three by all the classifiers when they are combined with CCEAFS (from 54). With the exception of SVM, the performance of all the classifiers has improved in terms of accuracy, precision, recall, and the F1 score. Accuracy using CCEAFS was 98.24% of all the classifiers except SVM, for which the accuracy was 97.65%—a small reduction compared to SVM without using FS.

Fig. 12 presents the performance of all classifiers on the diabetes dataset in terms of accuracy, precision, recall, and F1 score. The results indicate that the highest accuracy was obtained by LR without using FS, the second highest result was achieved by SVM, and the third highest by NB. After the classifiers were combined with CCEAFS, the number of features reduced from 8 to only 1 by J48, and to 2 by the other classifiers. The performance results show that without using

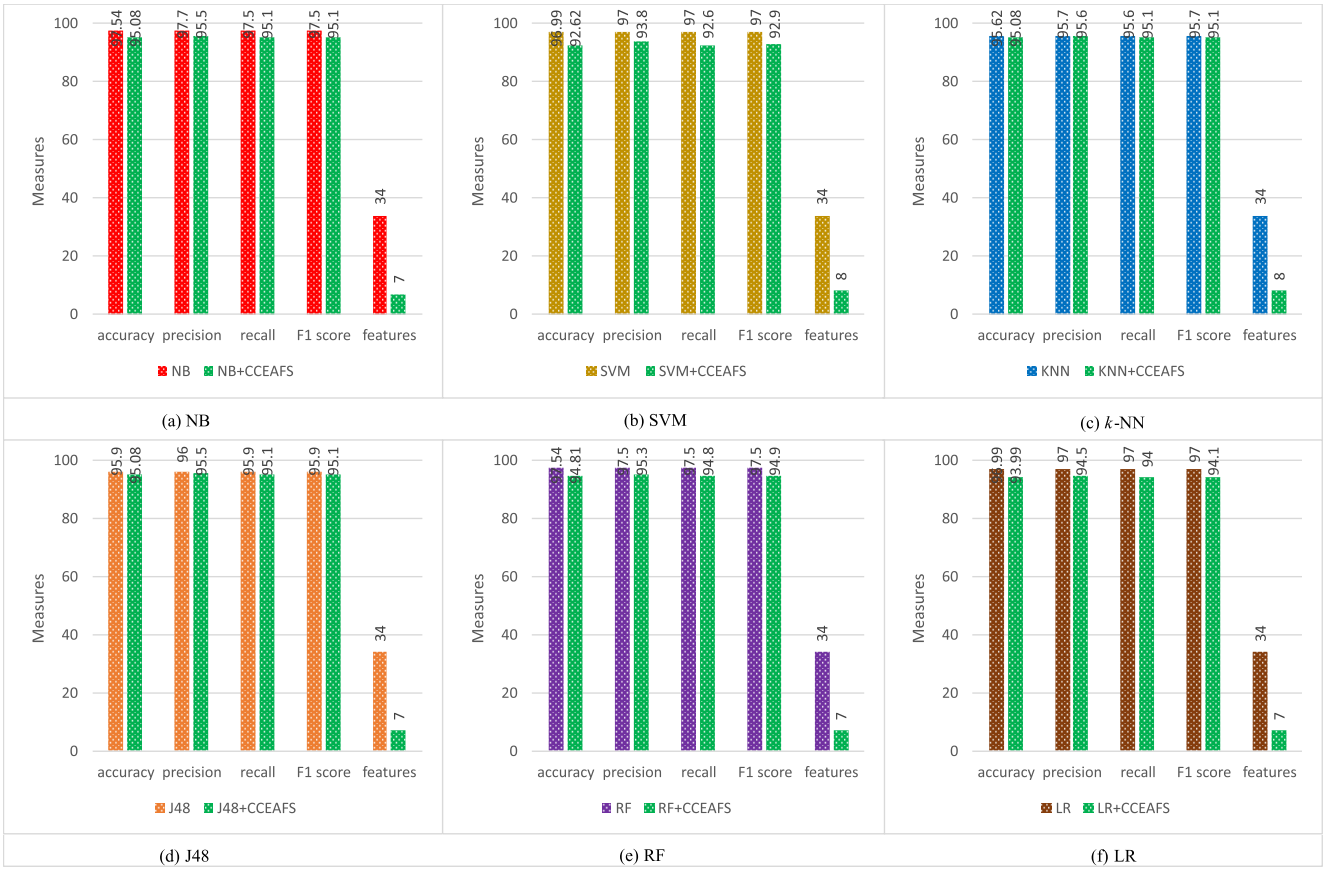

FIGURE 10. Performance evaluation of classifiers on the dermatology dataset.

FS, LR achieved the highest accuracy (77.21%), while when using FS, NB achieved the highest accuracy (76.56%).

Fig. 13 illustrates the performance of all classifiers on the musk dataset in terms of accuracy, precision, recall, F1 score, and features. The number of features was significantly reduced when CCEAFS was applied to the classifiers. The number of features reduced by J48+CCEAFS was only 14 (from 166), and all other classifiers achieved a reduction to between 16 to 25. The results indicate that the highest accuracy can be achieved by RF using all features in the dataset (97.34%), while NB achieved the lowest accuracy (84.04%). With the exception of NB, all classifiers, performance decreased slightly, for some more than for others. Although the performance in terms of all measures of k-NN, J48, and RF without FS and with CCEAFS have not been decreased significantly, these measures have dropped substantially for SVM, and LR when using CCEAFS. However, NB+CCEAFS has achieved higher performance except precision compared to NB without using FS.

Fig. 14 illustrates the performance of the NB classifier on the QSAR oral toxicity dataset in terms of accuracy, precision, recall, F1 score, and features. The number of features was significantly reduced when CCEAFS was applied to the classifiers. The feature reduction rate was 80.37%, which clearly indicates that without using FS, there is overfitting

TABLE 7. Summary of results for the wisconsin breast cancer dataset.

Classifiers	precision (%)	recall (%)	F1 score (%)	accuracy (%)	no. of features
NB	92.80	92.80	92.80	92.79	30
NB+CCEAFS	**93.10**	**93.10**	**93.10**	**93.15**	**2**
SVM	97.70	97.70	97.70	97.72	30
SVM+CCEAFS	93.00	92.80	92.70	92.79	**3**
k-NN	95.40	95.40	95.40	95.43	30
k-NN+CCEAFS	92.40	92.40	92.40	92.44	**3**
J48	93.00	93.00	93.00	92.97	30
J48+CCEAFS	91.20	91.00	90.90	91.04	**1**
RF	96.10	96.10	96.10	96.13	30
RF+CCEAFS	93.30	93.30	93.30	93.32	**2**
LR	95.00	94.90	94.90	94.90	30
LR+CCEAFS	92.40	92.40	92.40	92.44	**2**

with all features. Except the precision value, the performance of FS with NB has been greatly improved. The classification accuracy has been improved by 10.04%.

The experimental results for all of the datasets used in this paper are summarised in Table 7–12. The following points can be listed from the summarised results of all datasets:

Wisconsin Breast Cancer Dataset (Table 7)

- Without FS, in other words, using the full training dataset, the SVM classifier achieves the highest accuracy. Lower performance have been achieved by NB and

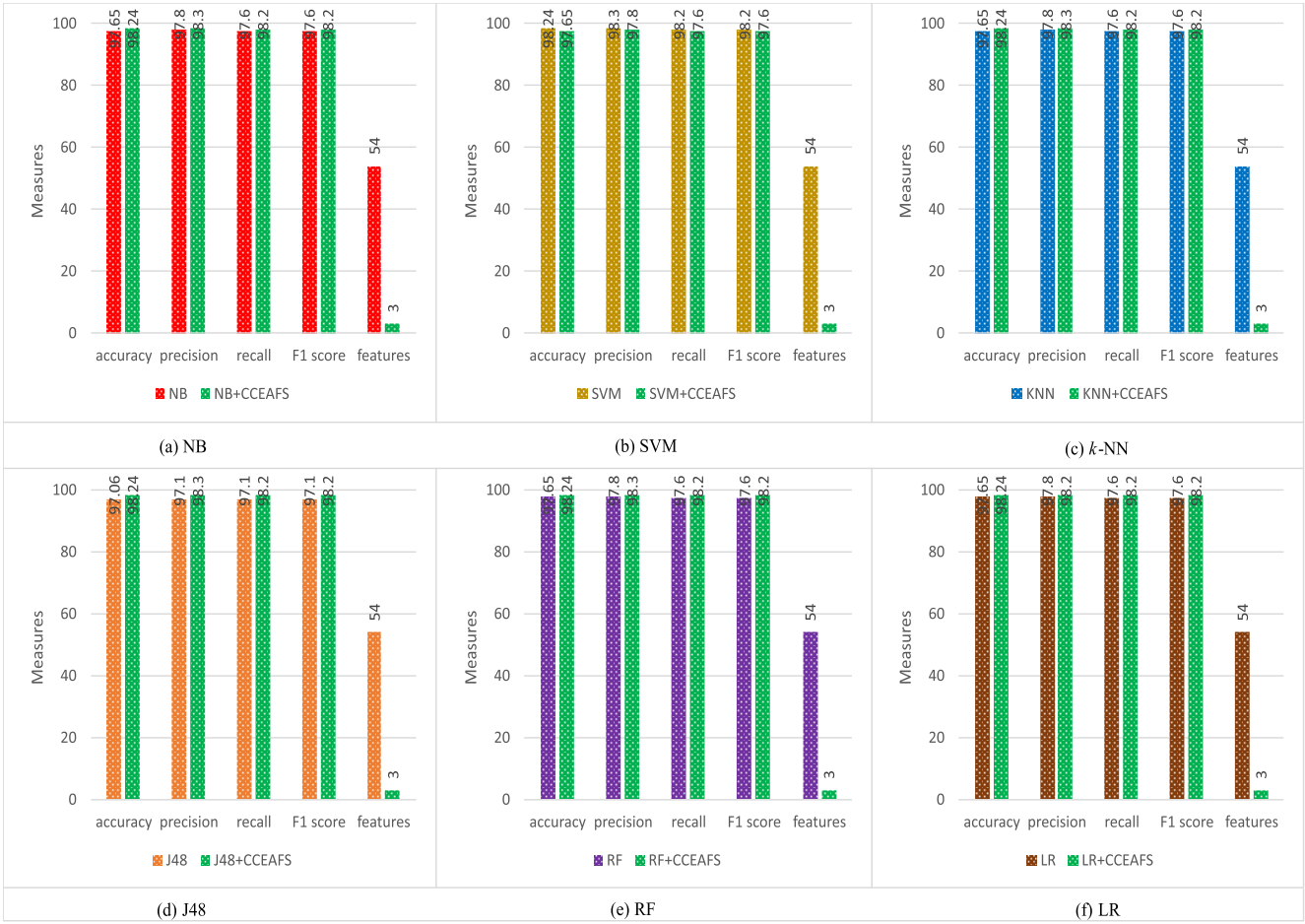

FIGURE 11. Performance evaluation of classifiers on the divorce dataset.

TABLE 8. Summary of results for the dermatology dataset.

Classifiers	precision (%)	recall (%)	F1 score (%)	accuracy (%)	no. of features
NB	97.70	97.50	97.50	97.54	34
NB+CCEAFS	95.50	95.10	95.10	95.08	**7**
SVM	97.00	97.00	97.00	96.99	34
SVM+CCEAFS	93.80	92.60	92.90	92.62	**8**
k-NN	95.70	95.60	95.70	95.62	34
k-NN+CCEAFS	95.60	95.10	95.10	95.08	**8**
J48	96.00	95.90	95.90	95.90	34
J48+CCEAFS	95.50	95.10	95.10	95.08	**7**
RF	97.50	97.50	97.50	97.54	34
RF+CCEAFS	95.30	94.80	94.90	94.81	**7**
LR	97.00	97.00	97.00	96.99	34
LR+CCEAFS	94.50	94.00	94.10	93.99	**7**

TABLE 9. Summary of results for the divorce dataset.

Classifiers	precision (%)	recall (%)	F1 score (%)	accuracy (%)	no. of features
NB	97.80	97.60	97.60	97.65	54
NB+CCEAFS	**98.30**	**98.20**	**98.20**	**98.24**	**3**
SVM	98.30	98.20	98.20	98.24	54
SVM+CCEAFS	97.80	97.60	97.60	95.24	**3**
k-NN	97.80	97.60	97.60	97.65	54
k-NN+CCEAFS	**98.30**	**98.20**	**98.20**	**98.24**	**3**
J48	97.10	97.10	97.10	97.06	54
J48+CCEAFS	**98.30**	**98.20**	**98.20**	**98.24**	**3**
RF	97.80	97.60	97.60	97.65	54
RF+CCEAFS	**98.30**	**98.20**	**98.20**	**98.24**	**3**
LR	97.80	97.60	97.60	97.65	54
LR+CCEAFS	**98.20**	**98.20**	**98.20**	**98.24**	**3**

J48 compared to other classifiers, and k-NN, RF, and LR achieved equally good performance.

- When the classifiers were combined with CCEAFS to reduce the dimensionality of the dataset, the performance of all the classifiers has been dropped except that of NB. RF and NB achieved the highest performance, and other classifiers achieved similar performance.

Dermatology Dataset (Table 8)

- The performance of k-NN and J48 classifiers is slightly less than other classifiers in terms of accuracy and other measures when the dataset is used with all of its features. The highest performance is achieved here by NB and RF classifiers together, whereas the lowest is achieved the by k-NN classifier.

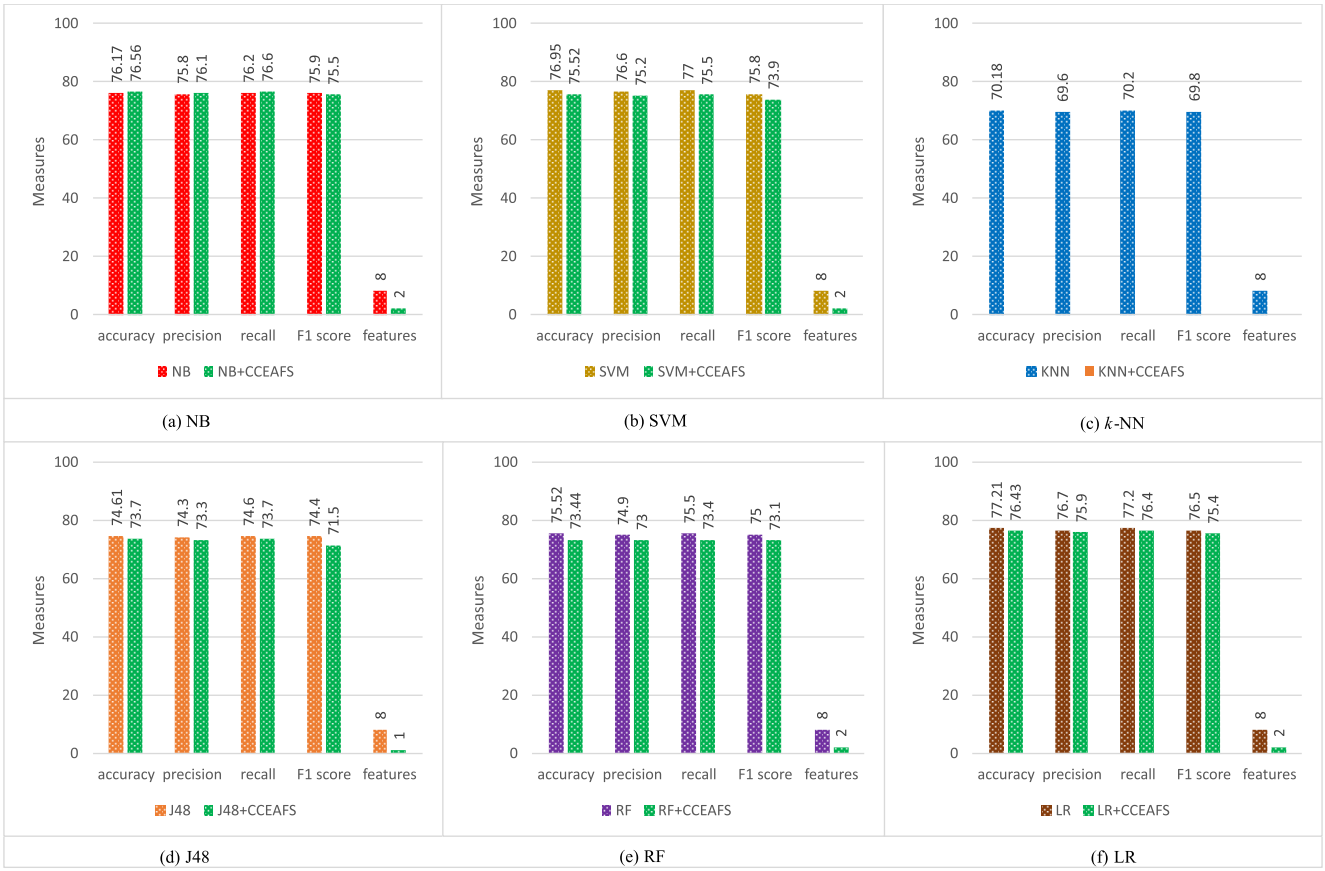

FIGURE 12. Performance evaluation of classifiers on the diabetes dataset.

TABLE 10. Summary of results for the diabetes dataset.

Classifiers	precision (%)	recall (%)	F1 score (%)	accuracy (%)	no. of features
NB	75.80	76.20	75.90	76.17	8
NB+CCEAFS	**76.10**	**76.60**	75.50	**76.56**	**2**
SVM	76.60	77.00	75.80	76.95	8
SVM+CCEAFS	75.20	75.50	73.90	75.52	**2**
k-NN	69.60	70.20	69.80	70.18	8
k-NN+CCEAFS	–	–	–	–	–
J48	74.30	74.60	74.40	74.61	8
J48+CCEAFS	73.30	73.70	71.50	73.70	**1**
RF	74.90	75.50	75.00	75.52	8
RF+CCEAFS	73.00	73.40	73.10	73.44	**2**
LR	76.70	77.20	76.50	77.21	8
LR+CCEAFS	75.90	76.40	75.40	76.43	**2**

TABLE 11. Summary of results for the Musk dataset.

Classifiers	precision (%)	recall (%)	F1 score (%)	accuracy (%)	no. of features
NB	87.70	84.00	85.30	84.04	166
NB+CCEAFS	85.20	**85.10**	85.20	**85.13**	**25**
SVM	94.80	94.80	94.50	94.82	166
SVM+CCEAFS	91.30	91.70	91.00	91.71	**22**
k-NN	95.80	95.80	95.80	95.76	166
k-NN+CCEAFS	94.80	94.80	94.80	94.83	**18**
J48	96.80	96.80	96.80	96.85	166
J48+CCEAFS	95.50	95.60	95.50	95.57	**14**
RF	98.00	97.90	97.90	97.34	166
RF+CCEAFS	97.00	97.10	97.00	97.06	**16**
LR	95.20	95.30	95.20	95.29	166
LR+CCEAFS	90.60	91.10	90.40	91.10	**23**

- To reduce the number of features in the dataset, when the CCEAFS is combined with different classifiers, the highest classification performance is achieved by NB, k-NN, and J48 classifiers, whereas the lowest one is by SVM; other classifiers are equally good in terms of all measures.

Divorce Dataset (Table 9)

- All classifiers performance is equally good except SVM in terms of all measures when the dataset is used with all features. SVM achieved a higher classification performance. It is noted that for all classifiers with an exception of SVM, the same results were achieved in all measures.

TABLE 12. Summary of results for the QSAR Oral Toxicity dataset.

Classifiers	precision (%)	recall (%)	F1 score (%)	accuracy (%)	no. of features
NB	90.70	79.80	83.70	79.78	1,024
NB+CCEAFS	90.20	**87.80**	**88.80**	**87.79**	**201**

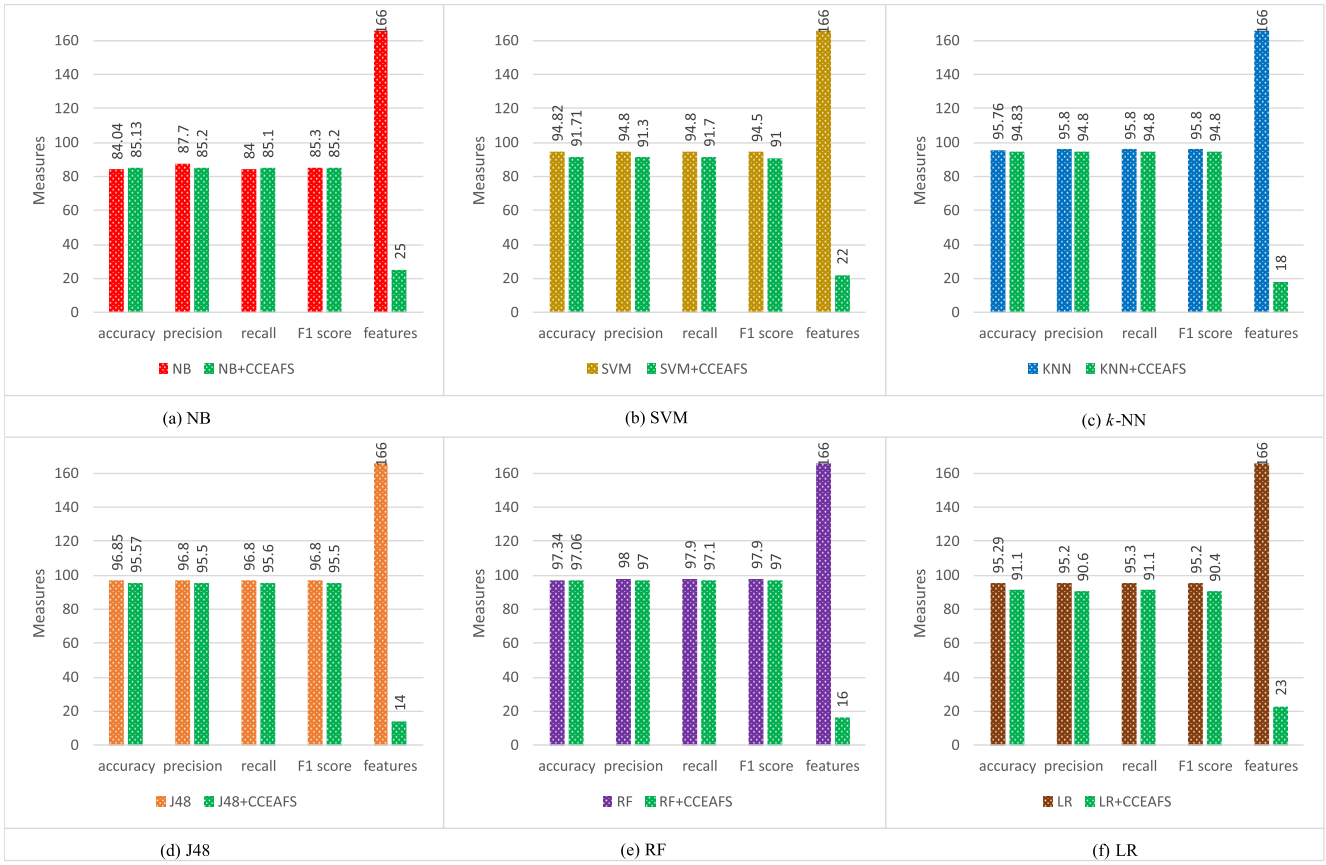

FIGURE 13. Performance evaluation of classifiers on the musk dataset.

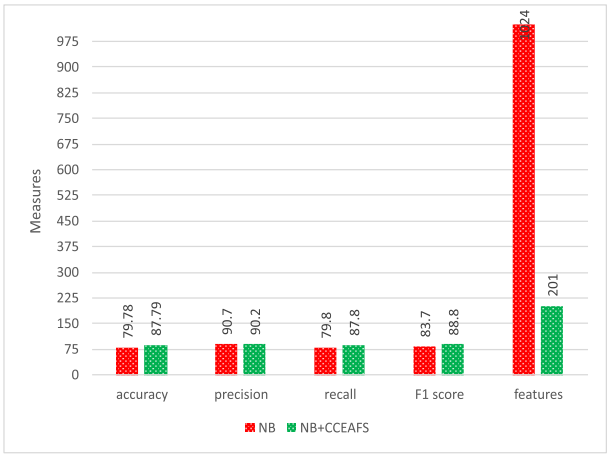

FIGURE 14. Performance evaluation of the NB classifier on the QSAR oral toxicity dataset.

- An improvement of all performance measures was observed for all classifiers except SVM as the number of features is reduced by combining the classifiers with CCEAFS. Similar to the results seen without using FS, all the classifiers perform at a similar level in terms of all measures except SVM.

Diabetes Dataset (Table 10)

- Equally good performance is observed by all classifiers except k-NN using the full training dataset.
- With the feature reduction using CCEAFS, all the classifiers perform with a slightly reduction in performance except NB, where the performance measures are increased in most cases.

Musk Dataset (Table 11)

- For the musk dataset using all features, it is reported that except NB, all other classifiers performance is equally good and RF achieved the highest performance in all cases. The performance by NB is much less than other classifiers.
- When the number of features is reduced in combination with CCEAFS, similar performance is observed by all classifiers except NB, and a performance drop is noted for all classifiers except NB. Here, NB achieved the highest performance in terms of recall, and accuracy to its counterpart using all features. Though in most cases, the overall performance is slightly dropped by all classifiers, the number of feature reduction is significant, with about 85% reduction in the number of features in the dataset.

QSAR Oral Toxicity Dataset (Table 12)

- For the QSAR oral toxicity dataset using all features, it can be observed that the NB classifier performs much better with than without using FS.

- Based on the performance of NB with CCEAFS, it is expected that CCEAFS, together with other classifiers, will also perform better along with a higher rate of dimensionality reduction.

According to the experiments, when the dimensionality of a dataset is low (as seen, for example, with the diabetes dataset), performance does not decrease substantially with the application of CCEAFS. With the increased number of features in a dataset, neither classifier's performance degrades much. During the experiments the NB classifier performed better in all cases except for one dataset when used in combination with CCEAFS. Therefore, using NB with CCEAFS can be recommended to reduce the number of features in datasets with large samples and few features.

VI. CONCLUSION AND FUTURE WORK

In this paper, the effect of the cooperative co-evolutionary algorithm for feature selection has been analyzed on six different widely used machine learning classification algorithms, namely, naïve Bayes, support vector machine, k-nearest neighbor, J48, random forest, and logistic regression. To address the identified issues, a penalty-based wrapper objective function has been proposed to be used as the fitness function for cooperative co-evolution, which leads to algorithm termination. This function has been effective at reducing the number of features in the dataset without significantly degrading performance. The performance of the classifiers was presented, both with and without feature selection. When keeping all the features in the dataset, SVM performed best in most cases, and LR in some of the cases. However, when the CCEAFS is applied, in most cases, NB outperformed the other classifiers.

The effectiveness of the proposed feature selection framework using cooperative co-evolution has been evaluated using static decomposition, a genetic algorithm used as the optimizer, and $1+N$ collaboration to build the complete solution. CCEA performance can be improved by using a dynamic decomposition technique, differential evolution as an optimizer, and an improved collaboration technique. Apart from the methods used in cooperative co-evolution, the datasets used in this work are characterized by large samples and few features. As future work, the effectiveness of the proposed feature selection framework will be tested using datasets with a larger number of features and a low number of samples. Another aspect to be investigated is CCEA combined with more effective decomposition methods, optimizers, and collaboration techniques studied in the literature.

REFERENCES

[1] A. B. Rashid and T. Choudhury, "Knowledge management overview of feature selection problem in high-dimensional financial data: Cooperative co-evolution and map reduce perspectives," *Problems Perspect. Manage.*, vol. 17, no. 4, p. 340, 2019, doi: 10.21511/ppm.17(4).2019.28.

[2] B. Chakraborty and A. Kawamura, "A new penalty-based wrapper fitness function for feature subset selection with evolutionary algorithms," *J. Inf. Telecommun.*, vol. 2, no. 2, pp. 163–180, Apr. 2018, doi: 10.1080/24751839.2018.1423792.

[3] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *Proc. Sci. Inf. Conf.*, Aug. 2014, pp. 372–378, doi: 10.1109/SAI.2014.6918213.

[4] J. Miao and L. Niu, "A survey on feature selection," *Procedia Comput. Sci.*, vol. 91, pp. 919–926, Jan. 2016, doi: 10.1016/j.procs.2016.07.111.

[5] M. Dash and H. Liu, "Feature selection for classification," *Intell. Data Anal.*, vol. 1, no. 3, pp. 131–156, 1997, doi: 10.1016/S1088-467X(97)00008-5.

[6] Y. Liu, F. Tang, and Z. Zeng, "Feature selection based on dependency margin," *IEEE Trans. Cybern.*, vol. 45, no. 6, pp. 1209–1221, Jun. 2015, doi: 10.1109/TCYB.2014.2347372.

[7] C. Elkan, "Naive Bayesian learning," Dept. Comput. Sci. Eng., Univ. California, San Diego, San Diego, CA, USA, Tech. Rep. CS97-557, 1997.

[8] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[9] A. Mucherino, P. J. Papajorgji, and P. M. Pardalos, "K-nearest neighbor classification," in *Data Mining in Agriculture*. New York, NY, USA: Springer, 2009, pp. 83–106.

[10] Z. Xiaoliang, Y. Hongcan, W. Jian, and W. Shangzhuo, "Research and application of the improved algorithm C4.5 on decision tree," in *Proc. Int. Conf. Test Meas.*, vol. 2, 2009, pp. 184–187, doi: 10.1109/ICTM.2009.5413078.

[11] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[12] F. S. Kurnaz, I. Hoffmann, and P. Filzmoser, "Robust and sparse estimation methods for high-dimensional linear and logistic regression," *Chemometric Intell. Lab. Syst.*, vol. 172, pp. 211–222, Jan. 2018, doi: 10.1016/j.chemolab.2017.11.017.

[13] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna, R. Kaluri, D. S. Rajput, G. Srivastava, and T. Baker, "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020, doi: 10.1109/ACCESS.2020.2980942.

[14] W. Gao, L. Hu, and P. Zhang, "Feature redundancy term variation for mutual information-based feature selection," *Appl. Intell.*, vol. 50, pp. 1–17, Jan. 2020, doi: 10.1007/s10489-019-01597-z.

[15] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003.

[16] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, Aug. 2016, doi: 10.1109/TEVC.2015.2504420.

[17] A. Bommert, X. Sun, B. Bischl, J. Rahnenführer, and M. Lang, "Benchmark for filter methods for feature selection in high-dimensional classification data," *Comput. Statist. Data Anal.*, vol. 143, Mar. 2020, Art. no. 106839, doi: 10.1016/j.csda.2019.106839.

[18] M. U. Öziç and S. Özşen, "T-test feature ranking based 3D MR classification with VBM mask," in *Proc. 25th Signal Process. Commun. Appl. Conf. (SIU)*, May 2017, pp. 1–4, doi: 10.1109/SIU.2017.7960591.

[19] E. Hancer, B. Xue, and M. Zhang, "Differential evolution for filter feature selection based on information theory and feature ranking," *Knowl.-Based Syst.*, vol. 140, pp. 103–119, Jan. 2018, doi: 10.1016/j.knosys.2017.10.028.

[20] A. K. Shukla and D. Tripathi, "Detecting biomarkers from microarray data using distributed correlation based gene selection," *Genes Genomics*, vol. 42, pp. 1–17, Feb. 2020, doi: 10.1007/s13258-020-00916-w.

[21] G. John and R. Kohavi, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, 1997, doi: 10.1016/S0004-3702(97)00043-X.

[22] E. E. Bron, M. Smits, W. J. Niessen, and S. Klein, "Feature selection based on the SVM weight vector for classification of dementia," *IEEE J. Biomed. Health Informat.*, vol. 19, no. 5, pp. 1617–1626, Sep. 2015, doi: 10.1109/JBHI.2015.2432832.

[23] A. Wang, N. An, G. Chen, L. Li, and G. Alterovitz, "Accelerating wrapper-based feature selection with K-nearest-neighbor," *Knowl.-Based Syst.*, vol. 83, pp. 81–91, Jul. 2015, doi: 10.1016/j.knosys.2015.03.009.

[24] S. Maldonado and J. López, "Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification," *Appl. Soft Comput.*, vol. 67, pp. 94–105, Jun. 2018, doi: 10.1016/j.asoc.2018.02.051.

[25] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc., B, Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.

[26] G. Biau, B. Cadre, and L. Rouvière, "Accelerated gradient boosting," *Mach. Learn.*, vol. 108, no. 6, pp. 971–992, Jun. 2019, doi: 10.1007/s10994-019-05787-1.

[27] C. J. Tan, C. P. Lim, and Y. Cheah, "A multi-objective evolutionary algorithm-based ensemble optimizer for feature selection and classification with neural network models," *Neurocomputing*, vol. 125, pp. 217–228, Feb. 2014, doi: 10.1016/j.neucom.2012.12.057.

[28] F. Moslehi and A. Haeri, "A novel hybrid wrapper–filter approach based on genetic algorithm, particle swarm optimization for feature subset selection," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 3, pp. 1105–1127, Mar. 2020, doi: 10.1007/s12652-019-01364-5.

[29] O. Soufan, D. Kleftogiannis, P. Kalnis, and V. B. Bajic, "DWFS: A wrapper feature selection tool based on a parallel genetic algorithm," *PLoS ONE*, vol. 10, no. 2, Feb. 2015, Art. no. e0117988, doi: 10.1371/journal.pone.0117988.

[30] K. Nag and N. R. Pal, "A multiobjective genetic programming-based ensemble for simultaneous feature selection and classification," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 499–510, Feb. 2016, doi: 10.1109/TCYB.2015.2404806.

[31] J. Tian, M. Li, and F. Chen, "Dual-population based coevolutionary algorithm for designing RBFNN with feature selection," *Expert Syst. Appl.*, vol. 37, no. 10, pp. 6904–6918, Oct. 2010, doi: 10.1016/j.eswa.2010.03.031.

[32] X.-F. Song, Y. Zhang, Y.-N. Guo, X.-Y. Sun, and Y.-L. Wang, "Variable-size cooperative coevolutionary particle swarm optimization for feature selection on high-dimensional data," *IEEE Trans. Evol. Comput.*, early access, Jan. 22, 2020, doi: 10.1109/TEVC.2020.2968743.

[33] S. Kashef and H. Nezamabadi-Pour, "An advanced ACO algorithm for feature subset selection," *Neurocomputing*, vol. 147, pp. 271–279, Jan. 2015, doi: 10.1016/j.neucom.2014.06.067.

[34] A. K. Shukla, P. Singh, and M. Vardhan, "Gene selection for cancer types classification using novel hybrid metaheuristics approach," *Swarm Evol. Comput.*, vol. 54, May 2020, Art. no. 100661, doi: 10.1016/j.swevo.2020.100661.

[35] A. K. Shukla, P. Singh, and M. Vardhan, "A new hybrid feature subset selection framework based on binary genetic algorithm and information theory," *Int. J. Comput. Intell. Appl.*, vol. 18, no. 3, Sep. 2019, Art. no. 1950020, doi: 10.1142/S1469026819500202.

[36] A. K. Shukla, "Feature selection inspired by human intelligence for improving classification accuracy of cancer types," *Comput. Intell.*, pp. 1–28, Jun. 2020, doi: 10.1111/coin.12341.

[37] E. Zorarpacı and S. A. Özel, "A hybrid approach of differential evolution and artificial bee colony for feature selection," *Expert Syst. Appl.*, vol. 62, pp. 91–103, Nov. 2016, doi: 10.1016/j.eswa.2016.06.004.

[38] M. Zhang, J. Ma, M. Gong, H. Li, and J. Liu, "Memetic algorithm based feature selection for hyperspectral images classification," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2017, pp. 495–502, doi: 10.1109/CEC.2017.7969352.

[39] M. Han and W. Ren, "Global mutual information-based feature selection approach using single-objective and multi-objective optimization," *Neurocomputing*, vol. 168, pp. 47–54, Nov. 2015, doi: 10.1016/j.neucom.2015.06.016.

[40] T. M. Hamdani, J.-M. Won, A. M. Alimi, and F. Karray, "Multi-objective feature selection with NSGA II," in *Proc. Int. Conf. Adapt. Natural Comput. Algorithms*. Berlin, Germany: Springer, 2007, pp. 240–247, doi: 10.1007/978-3-540-71618-1_27.

[41] Y. Yuan, H. Xu, and B. Wang, "An improved NSGA-III procedure for evolutionary many-objective optimization," in *Proc. Annu. Conf. Genet. Evol. Comput.*, 2014, pp. 661–668, doi: 10.1145/2576768.2598342.

[42] Y. P. Guo, X. B. Cao, Y. W. Xu, and Q. Hong, "Co-evolution based feature selection for pedestrian detection," in *Proc. IEEE Int. Conf. Control Automat.*, May 2007, pp. 2797–2801, doi: 10.1109/ICCA.2007.4376871.

[43] X. B. Cao, Y. W. Xu, C. X. Wei, and Y. P. Guo, "Feature subset selection based on co-evolution for pedestrian detection," *Trans. Inst. Meas. Control*, vol. 33, no. 7, pp. 867–879, Oct. 2011, doi: 10.1177/0142331209103041.

[44] J. Derrac, S. García, and F. Herrera, "A first study on the use of coevolutionary algorithms for instance and feature selection," in *Proc. Int. Conf. hybrid Artif. Intell. Syst.* Berlin, Germany: Springer, 2009, pp. 557–564, doi: 10.1007/978-3-642-02319-4_67.

[45] J. Derrac, S. García, and F. Herrera, "IFS-CoCo: Instance and feature selection based on cooperative coevolution with nearest neighbor rule," *Pattern Recognit.*, vol. 43, no. 6, pp. 2082–2105, Jun. 2010, doi: 10.1016/j.patcog.2009.12.012.

[46] Y. Wen and H. Xu, "A cooperative coevolution-based pittsburgh learning classifier system embedded with memetic feature selection," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2011, pp. 2415–2422, doi: 10.1109/CEC.2011.5949916.

[47] M. K. Ebrahimpour, H. Nezamabadi-Pour, and M. Eftekhari, "CCFS: A cooperating coevolution technique for large scale feature selection on microarray datasets," *Comput. Biol. Chem.*, vol. 73, pp. 171–178, Apr. 2018, doi: 10.1016/j.compbiolchem.2018.02.006.

[48] V. E. Christo, H. K. Nehemiah, J. Brighty, and A. Kannan, "Feature selection and instance selection from clinical datasets using co-operative co-evolution and classification using random forest," *IETE J. Res.*, pp. 1–14, 2020, doi: 10.1080/03772063.2020.1713917.

[49] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, "A review of microarray datasets and applied feature selection methods," *Inf. Sci.*, vol. 282, pp. 111–135, Oct. 2014, doi: 10.1016/j.ins.2014.05.042.

[50] M. Shi and S. Gao, "Reference sharing: A new collaboration model for cooperative coevolution," *J. Heuristics*, vol. 23, no. 1, pp. 1–30, Feb. 2017, doi: 10.1007/s10732-016-9322-9.

[51] M. Bhattacharya, R. Islam, and J. Abawajy, "Evolutionary optimization: A big data perspective," *J. Netw. Comput. Appl.*, vol. 59, pp. 416–426, Jan. 2016, doi: 10.1016/j.jnca.2014.07.032.

[52] V. Stanovov, C. Brester, M. Kolehmainen, and O. Semenkina, "Why don't you use evolutionary algorithms in big data?" *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 173, Feb. 2017, Art. no. 012020, doi: 10.1088/1757-899x/173/1/012020.

[53] M. A. Potter and K. A. De Jong, "A cooperative coevolutionary approach to function optimization," in *Proc. Int. Conf. Parallel Problem Solving From Nature*. Berlin, Germany: Springer, 1994, pp. 249–257, doi: 10.1007/3-540-58484-6_269.

[54] M. A. Potter and K. A. De Jong, "Evolving neural networks with collaborative species," in *Proc. Summer Comput. Simulation Conf.* San Diego, CA, USA: Society For Computer Simulation, 1995, pp. 340–345.

[55] H. Juillé and J. B. Pollack, "Co-evolving intertwined spirals," in *Proc. 5th Annu. Conf. Evol. Program.*, 1996, pp. 1–8.

[56] A. Bucci and J. B. Pollack, "On identifying global optima in cooperative coevolution," in *Proc. 7th Annu. Conf. Genet. Evol. Comput.*, 2005, pp. 539–544, doi: 10.1145/1068009.1068098.

[57] M. N. Omidvar, X. Li, Y. Mei, and X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 378–393, Jun. 2014, doi: 10.1109/TEVC.2013.2281543.

[58] Z. Yang, K. Tang, and X. Yao, "Large scale evolutionary optimization using cooperative coevolution," *Inf. Sci.*, vol. 178, no. 15, pp. 2985–2999, Aug. 2008, doi: 10.1016/j.ins.2008.02.017.

[59] M. N. Omidvar, M. Yang, Y. Mei, X. Li, and X. Yao, "DG2: A faster and more accurate differential grouping for large-scale black-box optimization," *IEEE Trans. Evol. Comput.*, vol. 21, no. 6, pp. 929–942, Dec. 2017, doi: 10.1109/TEVC.2017.2694221.

[60] M. A. Potter, "The design and analysis of a computational model of cooperative coevolution," Ph.D. dissertation, George Mason Univ., Fairfax, VA, USA, 1997.

[61] R. P. Wiegand, "An analysis of cooperative coevolutionary algorithms," Ph.D. dissertation, George Mason Univ., Fairfax, VA, USA, 2003.

[62] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997, doi: 10.1023/A:1008202821328.

[63] M. A. Potter and K. A. D. Jong, "Cooperative coevolution: An architecture for evolving coadapted subcomponents," *Evol. Comput.*, vol. 8, no. 1, pp. 1–29, Mar. 2000, doi: 10.1162/106365600568086.

[64] M. Banerjee, S. Mitra, and H. Banka, "Evolutionary rough feature selection in gene expression data," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 4, pp. 622–632, Jul. 2007, doi: 10.1109/TSMCC.2007.897498.

[65] W. N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," *Proc. SPIE*, vol. 1905, pp. 861–870, Jul. 1993, doi: 10.1117/12.148698.

[66] G. Demiroz, H. Govenir, and N. Ilter, "Learning differential diagnosis of erythemato-squamous diseases using voting feature intervals," *Artif. Intell. Med.*, vol. 13, no. 3, pp. 147–165, 1998, doi: 10.1016/s0933-3657(98)00028-1.

[67] M. K. Yöntem, K. Adem, T. I. İhan, and S. Kılıçarslan, "Divorce prediction using correlation based feature selection and artificial neural networks," *Nevşehir Hacı Bektaş Veli Üniversitesi SBE Dergisi*, vol. 9, no. 1, pp. 259–273, 2019.

[68] J. W. Smith, J. Everhart, W. Dickson, W. Knowler, and R. Johannes, "Using the ADAP learning algorithm to forecast the onset of diabetes mellitus," in *Proc. Annu. Symp. Comput. Appl. Med. Care*. Bethesda, MD, USA: American Medical Informatics Association, 1988, p. 261.
[69] T. G. Dietterich, A. N. Jain, R. H. Lathrop, and T. Lozano-Perez, "A comparison of dynamic reposing and tangent distance for drug activity prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 216–223.
[70] D. Ballabio, F. Grisoni, V. Consonni, and R. Todeschini, "Integrated QSAR models to predict acute oral systemic toxicity," *Mol. Informat.*, vol. 38, nos. 8–9, Aug. 2019, Art. no. 1800124, doi: 10.1002/minf.201800124.
[71] K. Yu, W. Shi, and N. Santoro, "Designing a streaming algorithm for outlier detection in data mining—An incrementa approach," *Sensors*, vol. 20, no. 5, p. 1261, Feb. 2020, doi: 10.3390/s20051261.
[72] T. Kenter, K. Balog, and M. de Rijke, "Evaluating document filtering systems over time," *Inf. Process. Manage.*, vol. 51, no. 6, pp. 791–808, Nov. 2015, doi: 10.1016/j.ipm.2015.03.005.
[73] K. L. Devi, P. Subathra, and P. Kumar, "Performance evaluation of sentiment classification using query strategies in a pool based active learning scenario," in *Computational Intelligence, Cyber Security and Computational Models*. Singapore: Springer, 2016, pp. 65–75, doi: 10.1007/978-981-10-0251-9_8.
[74] M. Mondal, R. Semwal, U. Raj, I. Aier, and P. K. Varadwaj, "An entropy-based classification of breast cancerous genes using microarray data," *Neural Comput. Appl.*, vol. 32, pp. 1–8, Nov. 2018, doi: 10.1007/s00521-018-3864-8.
[75] S. M. Alqahtani and R. John, "A comparative analysis of different classification techniques for cloud intrusion detection systems' alerts and fuzzy classifiers," in *Proc. Comput. Conf.*, Jul. 2017, pp. 406–415, doi: 10.1109/SAI.2017.8252132.
[76] M. H. Montoril, W. Chang, and B. Vidakovic, "Wavelet-based estimation of generalized discriminant functions," *Sankhya B*, vol. 81, no. 2, pp. 318–349, Dec. 2019, doi: 10.1007/s13571-018-0158-1.

A. N. M. BAZLUR RASHID received the B.Sc. degree in computer science and engineering from the Rajshahi University of Engineering and Technology, Bangladesh, in 2004, and the M.Sc. degree in information and communication technology from the Bangladesh University of Engineering and Technology, in 2010. He is currently pursuing the Ph.D. degree with the School of Science, Edith Cowan University, Australia. From 2005 to 2010, he has served in different organizations in different roles, such as database administrator and programmer. In 2010, he joined Comilla University, Bangladesh, as a Lecturer at the Department of Computer Science and Engineering. Since 2012, he has been an Assistant Professor (Computer) with the Bangladesh University of Textiles. He is the author of a number of conference and journal articles. His research interests include evolutionary computation, machine learning, big data optimization, data science, knowledge discovery, and decision support systems.

MOHIUDDIN AHMED received the Ph.D. degree from UNSW Australia. He is currently a Lecturer with the Academic Centre for Cyber Security Excellence, Edith Cowan University, Australia. He has published a number of journals and conferences papers in reputed venues of computer science and has edited a book on *Data Analytics* (CRC Press, USA). His research interests include big data mining, machine learning, and cybersecurity. He has made practical and theoretical contribution for data summarization for network traffic analysis. His research also has a high impact on critical infrastructure protection (SCADA systems and Smart Grid), information security against DoS attacks, and complicated health data (heart disease, nutrition) analysis. He is also the Cyber Security Editorial Advisory Board member at the Cambridge Scholars Publishing Group, U.K. He is also an Associate Editor of the *International Journal of Computers and Applications* (Taylor & Francis), U.K.

LESLIE F. SIKOS is currently a Computer Scientist specializing in network forensics and cybersecurity applications powered by artificial intelligence and data science. He has industry experience in data center and cloud infrastructures, cyberthreat prevention and mitigation, and firewall management. He regularly works on cybersecurity research projects, and collaborates with the Defence Science and Technology Group of the Australian Government, CSIRO's Data61, and the Cyber Security Collaborative Research Centre. He is a Reviewer of academic journals, such as *Computers & Security* and the IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, and the Chair Session at international conferences, and regularly edits books, on AI in cybersecurity. He holds professional certificates, and is a member of the IEEE Computer Society Technical Committee on Security and Privacy, and a Founding member of the IEEE Special Interest Group on Big Data for Cybersecurity and Privacy.

PAUL HASKELL-DOWLAND (Senior Member, IEEE) is currently an Associate Professor and the Associate Dean for Computing and Security with the School of Science, Edith Cowan University. He is also an Associate Member of the Centre for Security, Communications & Network Research, Plymouth University, U.K. He has delivered keynotes, invited presentations, workshops, professional development/training, and seminars across the world for audiences, including RSA Security, Sri Lanka CERT, ITU, and IEEE. He has appeared on local and national media (newspaper, radio, and TV) commenting on current cyber issues as well as contributions through articles published in The Conversation. He is the author of over 80 articles in refereed international journals and conference proceedings and edited 29 proceedings. He has more than 20 years of experience in cyber security research and education in both the U.K. and Australia. He is the Working Group Coordinator and the ACS/Australian Country Member Representative to the International Federation for Information Processing (IFIP) Technical Committee 11 (TC11 - Security and Privacy Protection in Information Processing Systems), and the Secretary to IFIP Working Group 11.1 (Information Security Management). He is a member of the ACS Cyber Security Committee, a Senior Member of the ACS/Certified Professional, a Fellow of the Higher Education Authority and BCS, and an Honorary Fellow of the Sir Alister Hardy Foundation for Ocean Science.